挑战性题目DSCT102：木板切割问题

问题描述：将一根长度为*L*的木板切割为*N*块，切割之后的木板长度为*l*1, *l*2, *...*, *lN*，即有。每次切割木板时候的开销为该块木板的长度。例如将长度为13的目标切割成6和7，那么切割开销为13。给定最终切割目标，请问最小的切割开销？例如，输入8 5 8，输出34。

数学描述：对输入的有限集合S={*a*1, *a*2,…, *aN*}，求得最优归并解。

思路分析[[1]](#footnote-1)与算法描述[[2]](#footnote-2)：

利用简单的数学知识可知，重复同样的次数，数据越小的段带来的总开销越小。因此，我们应该将木板的各个部分从小到大进行拼接。

假设我们将木板切割成N段，并且将这N段的值进行从小到大的排序。为了方便我们理解这样做得到的是最小开销的结果，我们将从逆向来思考这个问题：即不是将木板切割成各个部分，而是将切割完成的木板的各个部分拼接成原始的木板。容易知道的是，这两种做法当然是等价的。

由第一段中的叙述可知，当我们将木板由小到大排序后，我们将最短的两段木板依次进行拼接，得到一块新的木板；随后将这一新的木板与第三短的木板进行拼接，并以此类推。这样最后得到的总数值便是最小开销。

事实上，这样的做法就是遵循了霍夫曼树编码的方式，每次取数组中最小的两个值进行相加以得到最小开销。

由于无论是维护优先队列还是手动维护数组的有序性，都需要用到排序算法，故此处的时间复杂度将被开销，而又由于同时需要对数组进行循环处理，故最终的时间复杂度为。

1. 算法思路不但要给出解题的算法内涵，还应该分析该算法的时间复杂度。此页背面也可以答题，但不应续页。 [↑](#footnote-ref-1)
2. 代码的写作和测试建议使用GCC、G++等通用C/C++编译器进行编译以利于跨平台的性能测试。作业纸质版本和测试的可执行文件请提交给主管助教。测试样例命令格式为：DSCT102\_2018270103012.exe 8 5 8，输出结果样例为：34。 [↑](#footnote-ref-2)